4

Writing the Setup Program

In This Chapter. . . .

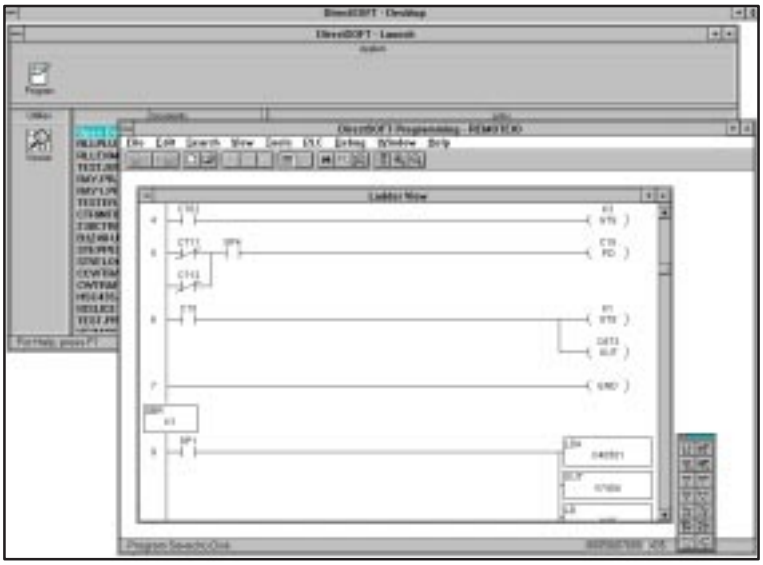
— Getting Started with Your Programming
— Writing Your Remote I/O Setup
— Special Relays used for Remote I/O
— How to Use the Special Relays

Getting Started with Your Programming

You can write your program using either a handheld programmer or PC loaded software such as ***Direct***SOFT. The examples that follow will show you how this is done using ***Direct***SOFT.

To get started, enter ***Direct***SOFT and carry out the normal ***Direct***SOFT setup procedures for communicating with your DL405 CPU. If you do not know how to do this, refer to your ***Direct***SOFT Manual. Chapter 11 of your DL405 User Manual also has a very good coverage of the basic commands available and examples of how the commands are used for writing general ladder logic. We will be showing you in this chapter only those commands that pertain to setting up your remote I/O initialization and its successful utilization.

First open ***Direct***SOFT from Windows and establish a link with your CPU. Then enter the Edit Mode for programming. You should now be looking at a screen similar to the one shown below:



The ***Direct***SOFT window shown above depicts a program that has already been written. Your window, of course, will be empty when you first enter it. The pages that follow will show you how to write each part of your initilization program.

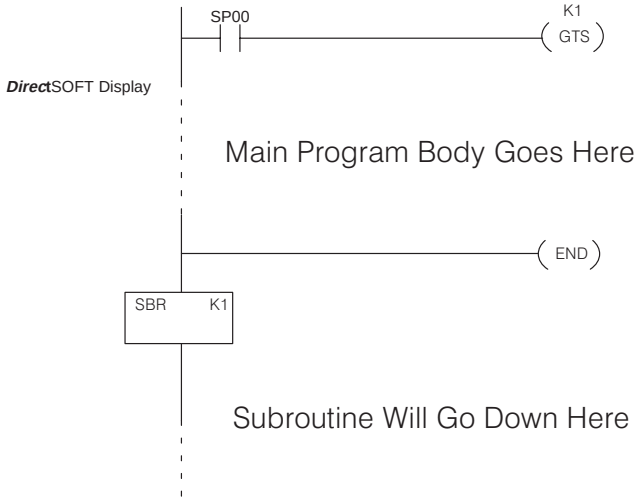
Writing Your Remote I/O Setup

Step 1: Decide How You Are Going to Call Your Program

Is your setup logic going to be in the main program body or is it going to be in a subroutine? If you have a DL430, the decision is made for you. The DL430 cannot handle the GTS command for calling a subroutine; and so, you have to write the code in the main body. The DL440, on the other hand, does include the GTS command.

A subroutine for your remote I/O setup has an advantage over writing the code into the program's main body. Some remote I/O setup logic becomes quite lengthy. By putting the setup in a subroutine, you don't have to scroll through extra logic during routine troubleshooting procedures. If you are using the DL440, we advise you to use a subroutine for your remote I/O initialization. Here's how:

Using the GTS Command for the DL440



NOTE: Set Retentive Ranges so that C670 – C737 and V7404 – V7477 are **not** retentive.

Step 2: Write the Setup Logic for Each Remote Base

Whether you choose to write the remote I/O setup program as a subroutine or as a part of the main program, the procedure is still the same. You have two things you have to do:

- Tell the CPU where to read and write the remote I/O points in memory. This is done with the use of “address pointers”.
- Tell the CPU how many points are located in each base.

You can use your worksheets to assist you. In the diagram below, you see how the starting addresses for the points in each remote base (from the tables in Appendix B) are mapped with the proper reserved memory pointers. The chart at the bottom of the page shows the pointer addresses. Notice that the number of points goes in the address immediately following the pointer for the start address. A combination of LDA and OUT commands are used to load and map the V40xxx address into the proper V74xx address. The LD and OUT commands are used to load the number of remote points for each remote base, by placing the number in the address immediately following its pointer. The chart at the bottom also shows the memory locations for storing the number of I/O points for each remote base.

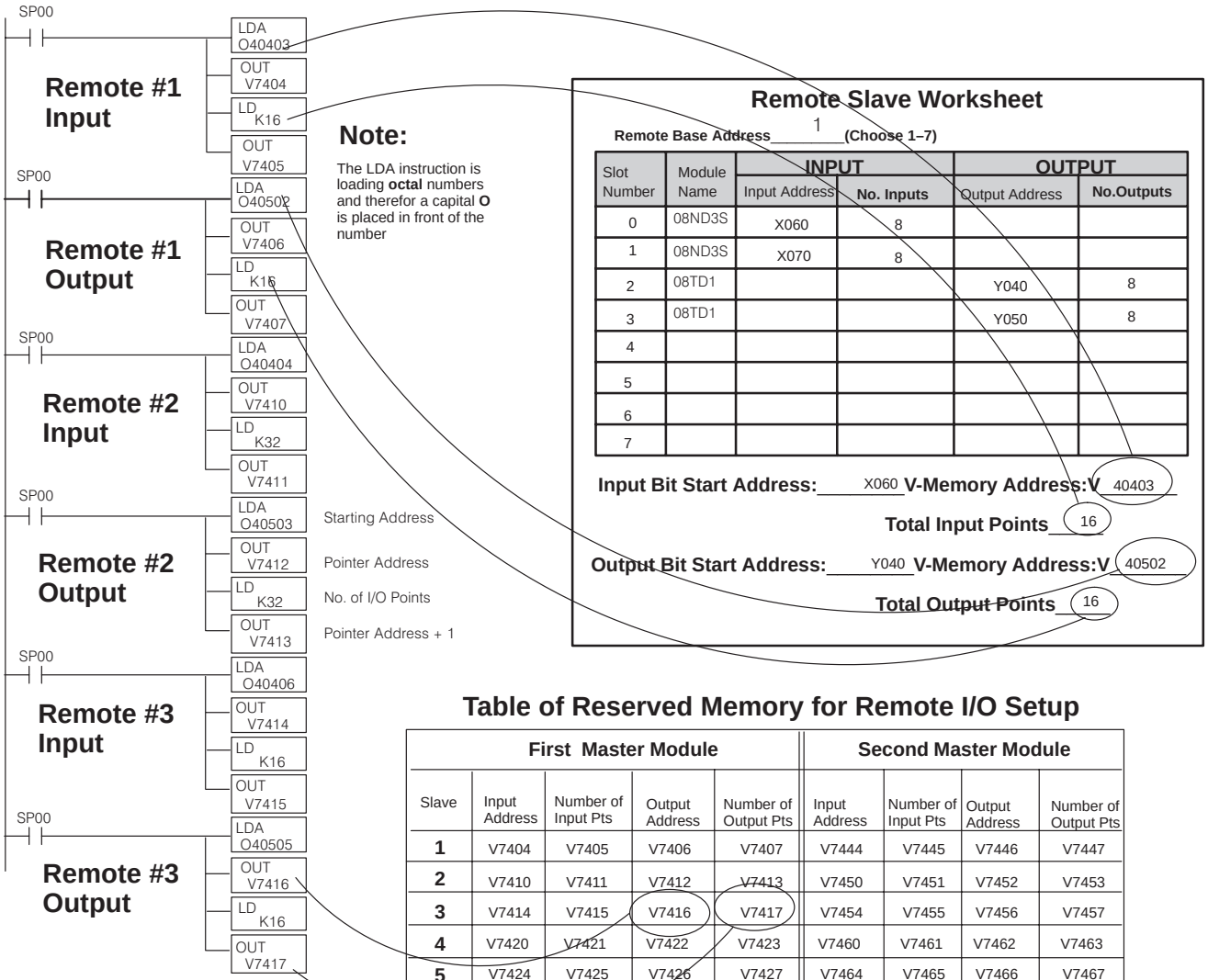


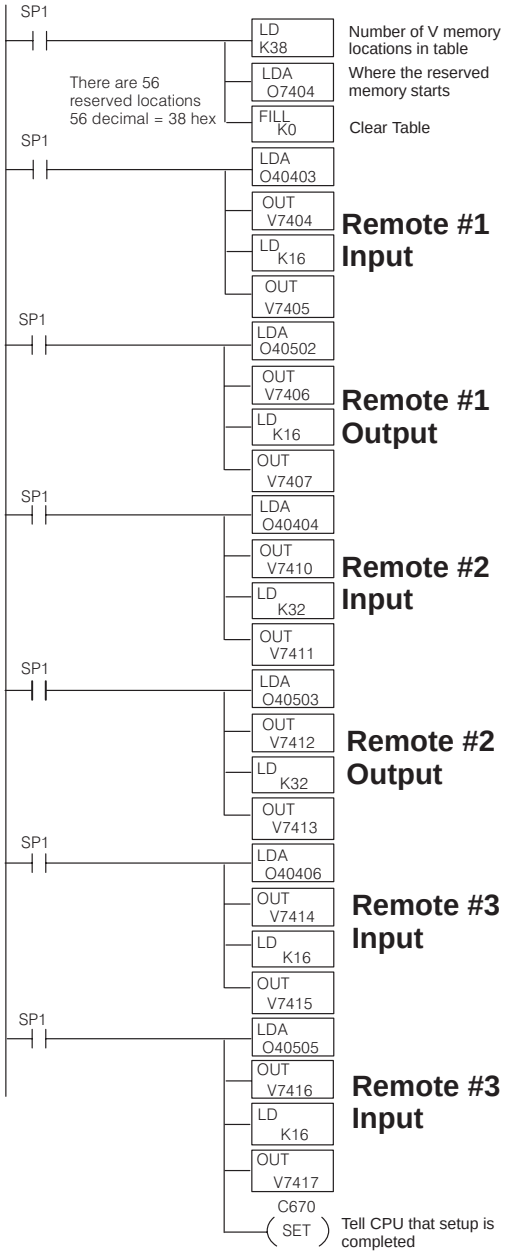
Table of Reserved Memory for Remote I/O Setup

	First Master Module				Second Master Module			
Slave	Input Address	Number of Input Pts	Output Address	Number of Output Pts	Input Address	Number of Input Pts	Output Address	Number of Output Pts
1	V7404	V7405	V7406	V7407	V7444	V7445	V7446	V7447
2	V7410	V7411	V7412	V7413	V7450	V7451	V7452	V7453
3	V7414	V7415	V7416	V7417	V7454	V7455	V7456	V7457
4	V7420	V7421	V7422	V7423	V7460	V7461	V7462	V7463
5	V7424	V7425	V7426	V7427	V7464	V7465	V7466	V7467
6	V7430	V7431	V7432	V7433	V7470	V7471	V7472	V7473
7	V7434	V7435	V7436	V7437	V7474	V7475	V7476	V7477

Tell the CPU That You Are Finished With the Setup

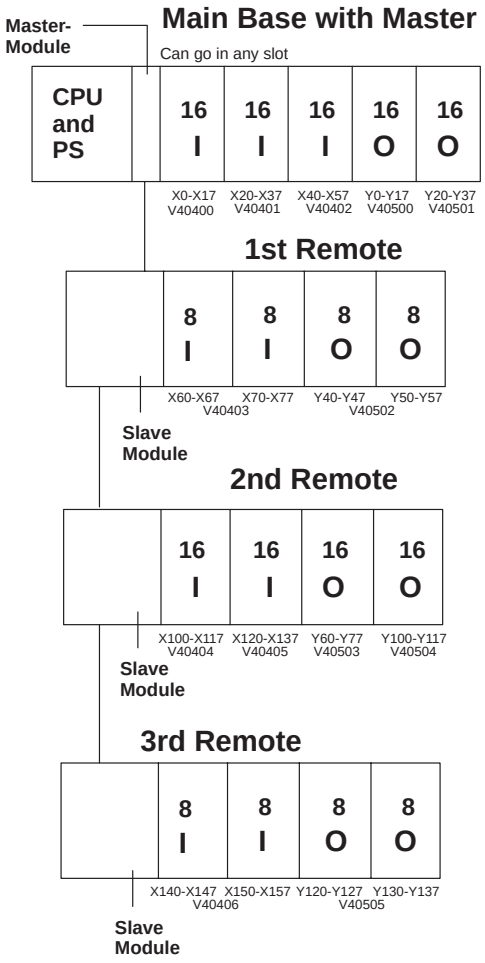
Once you have decided on the starting addresses and the reserved memory locations for each remote base, you have to zero out all of the reserved memory locations you are not going to use and then tell the CPU that you are finished with the setup. If you don't use the FILL command to insert zero's in the unused areas, the CPU will assume that every pointer address V7404 through V7477 is pointing to a read or write start address. This could cause problems. You may have garbage in these locations-–at the very least, its going to take up unnecessary scan time.

The easiest way to fill the unused memory locations with zeros is to do it first, before loading your table setup. Then you overwrite those memory locations used during your table setup, and everything else is zeros, as required. The example below approaches the problem this way.



EXAMPLE:
38.4 kBaud, D4-440

Step 1: Design the Remote I/O System



Remote Slave Worksheet

Remote Base Address 1 **(Choose 1–7)**

Slot Number	Module Name	INPUT		OUTPUT	
		Input Address	No. Inputs	Output Address	No.Outputs
0	08ND3S	X060	8		
1	08ND3S	X070	8		
2	08TD1			Y040	8
3	08TD1			Y050	8
4					
5					
6					
7					

Input Bit Start Address: X060 **V-Memory Address:V** 40403

Total Input Points 16

Output Bit Start Address: Y040 **V-Memory Address:V** 40502

Total Output Points 16

Remote Slave Worksheet

Remote Base Address 2 **(Choose 1–7)**

Slot Number	Module Name	INPUT		OUTPUT	
		Input Address	No. Inputs	Output Address	No.Outputs
0	16ND2	X100	16		
1	16ND2	X120	16		
2	16TD1			Y060	16
3	16TD1			Y100	16
4					
5					
6					
7					

ut Bit Start Address: X100 **V-Memory Address:V** 40404

Total Input Points 32

tput Bit Start Address: Y060 **V-Memory Address:V** 40503

Total Output Points 32

Remote Slave Worksheet

Remote Base Address 3 **(Choose 1–7)**

Slot Number	Module Name	INPUT		OUTPUT	
		Input Address	No. Inputs	Output Address	No.Outputs
0	08ND3S	X140	8		
1	08ND3S	X150	8		
2	08TD1			Y120	8
3	08TD1			Y130	8
4					
5					
6					
7					

Input Bit Start Address: X140 **V-Memory Address:V** 40406

Total Input Points 16

Output Bit Start Address: Y120 **V-Memory Address:V** 40505

Total Output Points 16

Note:

The Remote Slave Worksheet is found in Appendix A.

Step 2: Set the Hardware

Table for setting DIP switch

Position	1	2	3	4
Master	Always ON	ON=38.4kB OFF=19.2kB	Always OFF	Always OFF
Remote	Always OFF	ON=38.4kB OFF=19.2kB	Always OFF	Always OFF

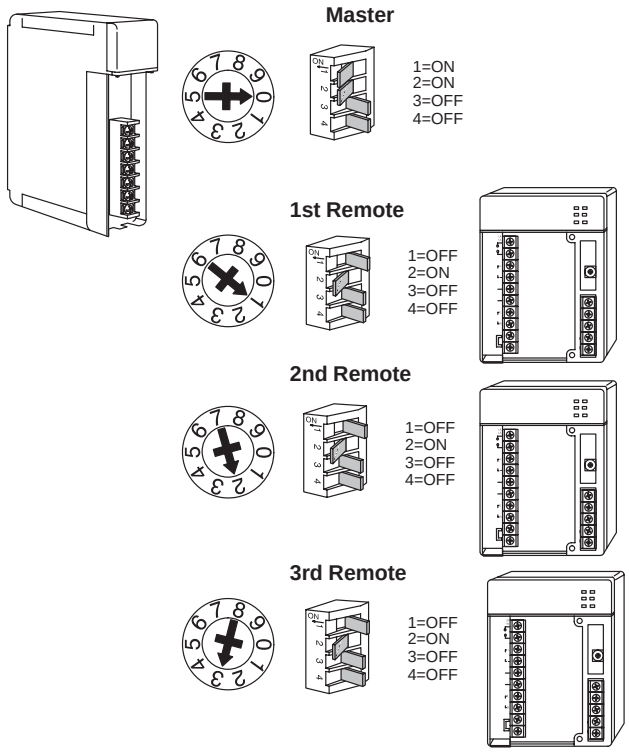


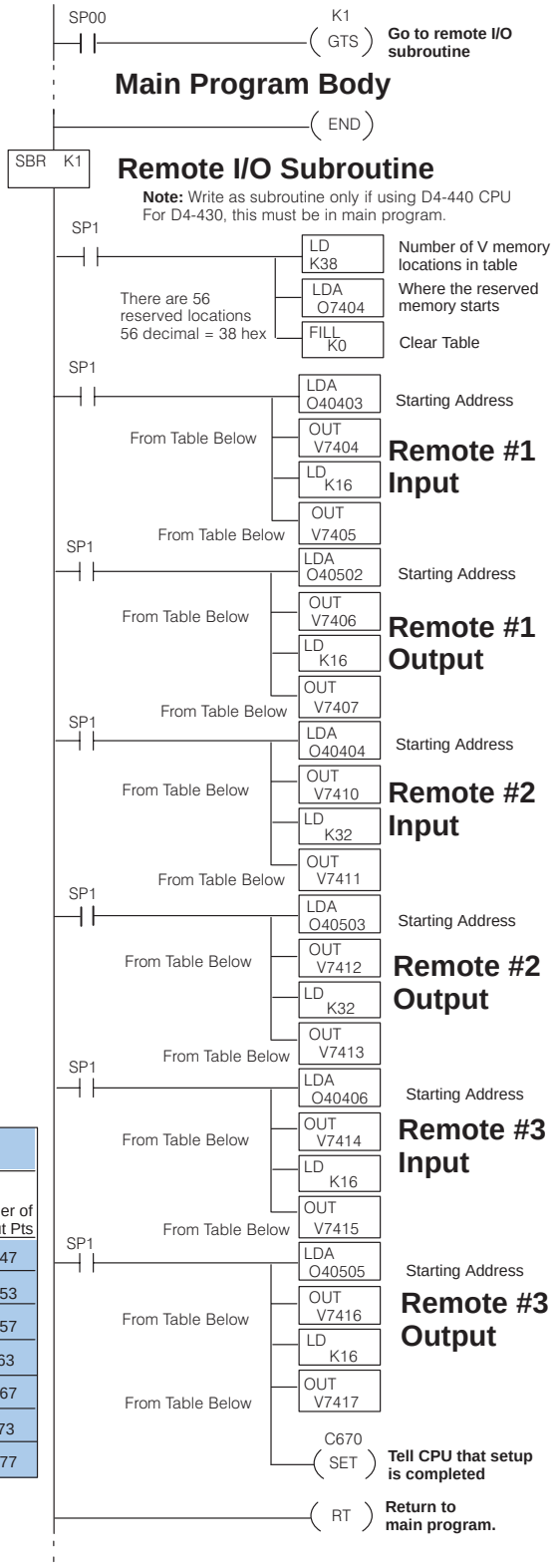
Table of Reserved Memory for Remote I/O Setup

	First Master Module				Second Master Module			
Slave	Input Address	Number of Input Pts	Output Address	Number of Output Pts	Input Address	Number of Input Pts	Output Address	Number of Output Pts
1	V7404	V7405	V7406	V7407	V7444	V7445	V7446	V7447
2	V7410	V7411	V7412	V7413	V7450	V7451	V7452	V7453
3	V7414	V7415	V7416	V7417	V7454	V7455	V7456	V7457
4	V7420	V7421	V7422	V7423	V7460	V7461	V7462	V7463
5	V7424	V7425	V7426	V7427	V7464	V7465	V7466	V7467
6	V7430	V7431	V7432	V7433	V7470	V7471	V7472	V7473
7	V7434	V7435	V7436	V7437	V7474	V7475	V7476	V7477

= unused memory for this example

Step 3: Write the Setup Program

RLL Program



Special Relays Used for Remote I/O

The remote I/O system has several relays that are used with your system. On the previous pages, you saw how C670 is used to tell the CPU that all of the mapping has taken place. Below is a complete list of all of these relays:

Function of Relay	First Master Relay (s)	Second Master Relay (s)	Description
Setup Complete (Mandatory)	C670	C674	These two relays are used to tell the CPU that your program has finished doing all of its remote I/O mapping. When finished, the CPU continues the rest of its scan cycle.
Locate Error	C700-C707	C720-C727	These relays are flags to let you know that a communication error has occurred. If set, there has been an error. This method of error detection helps locate the error. The last digit of the relay number indicates base unit. For example, C723 refers to the third slave unit of the second master. If it were C705, it would be indicating that the fifth slave unit of the first master module is not communicating.
I/O Status On Error (Save or Clear)	C671	C675	These two relays are for determining whether you want the remote I/O points to be set to zero when an error occurs, or whether you want to save the current I/O settings.
Restart But Ignore Part of System Causing Error	C673	C677	You may want to continue updating I/O data from remote I/O bases even if one of them has caused a communications error. These two relays allow you to take the bad base off line and to restart the system before the error is cleared.
Communications OK	C710-C717	C730-C737	These flags tell you if a particular base unit is ready for communication. The last digit of the relay number indicates the base unit. For example, C711 refers to the first slave unit of the first master. If it were C735, it would be indicating the communications status of the fifth slave unit of the second master module.

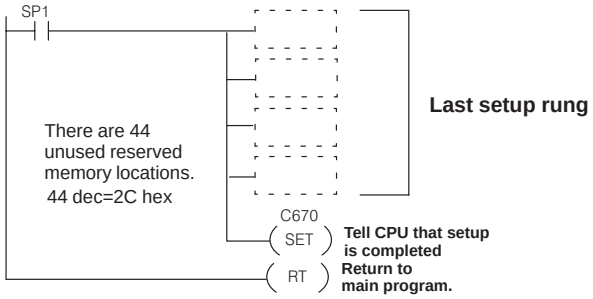
How to Use the Special Relays

Here are some example uses of these relays and an added explanation for each of the relays discussed on the previous page:

C670/C674: Setup Complete (Mandatory)

These are setup flags for **marking the end** of your ladder logic that sets up your remote I/O configuration. It should be the last rung of your setup. It should always follow your FILL command that zero's out all of the unused pointer addresses.

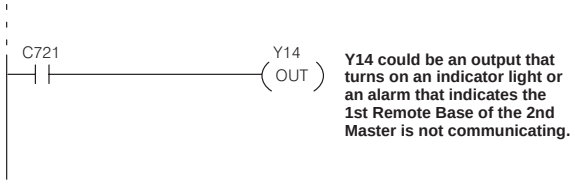
Example:



C700 to C707 and C720 to C727: Locate Communications Error (Optional)

C700 to C707 are assigned to the 1st Master. C720 to C727 are assigned to the 2nd Master. The last digit of these relays indicates the base unit number. Remember that the CPU base is always Base Unit #0. The remote bases can be any number 1 through 7. For example, C721 refers to the 2nd Master, 1st Remote Base. These relays will be set when there is a **communications error** between the respective master and slave assigned to the relay number.

Example:



C671/C675: I/O Status On Error (Optional)

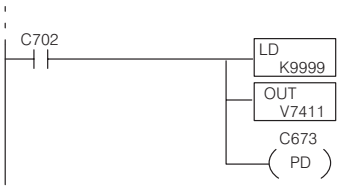
C671 is assigned to the 1st Master. C675 is assigned to the 2nd Master. When any master can't talk to one or more of its slaves, the "link" LED will illuminate on the affected module and the system will stop updating the remote I/O status in the CPU. You have several options at that point. One such option is either to **save the last known I/O status** that is in the CPU's memory image area, or to **write a zero to each point**. If these flags are OFF when the error occurs, all current I/O will be zeroed.

Example:

C673/C677 Error/Restart But Ignore Problem Area (Optional)

When a relay with C700 to C707 or C720 to C727 is set to indicate a communications error, you can use either or both C673 (for I/O belonging to the 1st master) and C677 (for I/O belonging to the 2nd master) as a method for having the CPU **skip the scanning** of the I/O register associated with a particular slave unit. Look in the Reserved Memory Table below to find the appropriate V74xx pointer address to match up with the approriate C7## relay. Both the relay and the pointers are specifically assigned to unique slave units.

Example:



The number 9999 loaded in the pointer address for Slave #2 of the 1st master will tell the CPU to ignore this slave unit during restart after an error. See previous page for proper use of C700 to C707 and C720 to C727.

	1st Master		2nd Master	
Slave	Relay	Address Pointer	Relay	Address Pointer
1	C701	V7405	C721	V7445
2	C702	V7411	C722	V7451
3	C703	V7415	C723	V7455
4	C704	V7421	C724	V7461
5	C705	V7425	C725	V7465
6	C706	V7431	C726	V7471
7	C707	V7435	C727	V7475

C710 to C717 and C730 to C737 Communications OK Status (Optional)

C710 to C717 are assigned to the 1st Master. C730 to C737 are assigned to the 2nd Master. The last digit of these relays refers to the base unit number. Remember that the CPU base is always Base Unit #0. The remote bases can be any number 1 through 7. For example C715 refers to the 1st Master, 5th Remote Base. These flags indicate that a particular slave unit is **ready for communcating data** over its twisted pair cable.

Example:

Y27 could be turning on an indicator light when the 5th Remote Base connected to the 1st Remote Master is ready for communications.

Example of RLL Using All the Special Relays